ECE 443/643 Lab 2

September 26, 2011

1 Cyclic Discrete Correlation

We had previously defined cross-correlation for continuous time energy signals as

$$\rho_{x,y}(\tau) = \int_{-\infty}^{\infty} x(t)y^*(t-\tau)dt$$

and for continuous time periodic power signals as

$$\rho_{x,y}(\tau) = \frac{1}{T}\int_{T} x(t)y^*(t-\tau)dt.$$

The analogous definition for periodic discrete time signals ($x[n] = x[n-N]$) is

$$\rho_{x,y}[\ell] = \sum_{n=0}^{N-1} x[n]y^*[n-\ell].$$

We can compute this with the discrete Fourier transform (DFT) just as we can with the continuous time versions with their respective Fourier transforms. We can also define correlation such that the arrays are not assumed to be periodic, but we cannot directly compute it with the DFT. Computing directly with the DFT is highly desirable because it is conceptually the same as the continuous time cases and fast because of the FFT. The spectral density is

$$S_{x,y}[k] = \mathrm{DFT}(\rho_{x,y}[\ell]) = X[k]Y^*[k].$$

This is easy to prove; it's just a matter of substituting in the definitions of the DFT and $\rho_{x,y}[\ell]$ and switching the order of summation; you should attempt it. It may be easiest to derive analogous time shift and conjugation rules for the DFT first.

2 Noise Rejection

The companion Matlab script for this lab generates a message m with 8 bits and a pulse consisting of a so-called *maximum length sequence*. Maximum length sequences are discrete sequences that have many nice properties, most important of which for this lab is the fact that they have very low autocorrelation except at $\ell = 0$. For each bit b of m, a pulse is multiplied by 1 if b is 1 and by -1 if b is 0. These multiplied pulses are concatenated together to form a transmitted signal x. x is corrupted by additive white Gaussian noise (AWGN) and is received as y. Can we recover m from y?

1. Plot p, its magnitude spectrum, and autocorrelation. Comment on these plots.
2. Plot n, its magnitude spectrum, and autocorrelation. Comment on these plots.
3. Run the full script several times (it will change the message and noise each time). With an SNR of 0.5, can you reliably recover the message by correlating y with the signal d? I.e. can you decide if the peaks from the matched filter indicate a 1 or 0 without error? Describe what correlating with d is doing.
4. Decrease the SNR to 0.1. Is it still reliable? How about 0.05? Find a noise power such that the detector sometimes fails and sometimes succeeds at recovering the message.

5. With your noise set to a value such that the communication is unreliable, try increasing the length of the pulse. You can do this by increasing the second argument of `mseq(2,a)`; the length of the pulse is $2^a - 1$. Why does this help?

6. Experiment with another type of pulse, e.g. a sinusoid or rectangle. Ensure that your pulse has the same energy as the maximum length sequence. Comment on why it performs better or worse.